

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平10-125794

(43)公開日 平成10年(1998)5月15日

(51) Int.Cl.<sup>8</sup>

識別記号

FI

H O 1 L 21/82

H0 1 L 21/82

T

G O I R 31/28

G O I R 31/28

F

G O 6 F 17/50

G O 6 F 15/60

670

H 0 1 L 21/82

C

審査請求 有 請求項の数 9 FD (全 19 頁)

(21)出願番号 特願平8-297599

(22)出願日 平成8年(1996)10月18日

(71)出願人 000004237

日本電気株式会社

東京都港区芝五丁目7番1号

(72)発明者 眞田 克

東京都港区芝五丁目7番1号 日本電気株式会社内

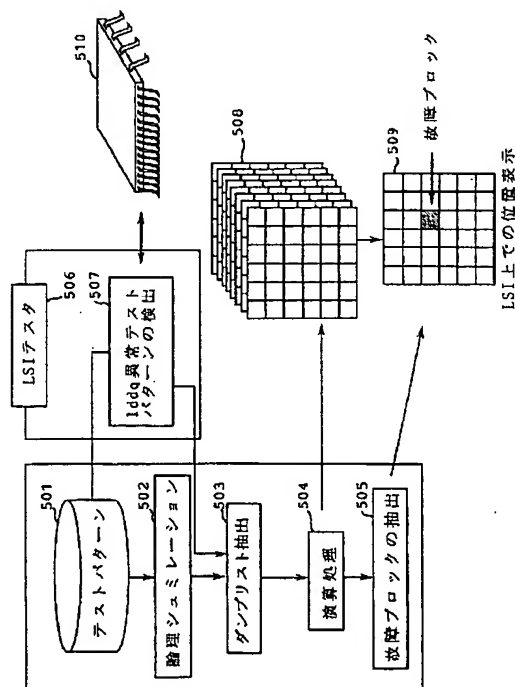
(74)代理人 弁理士 加藤 朝道

(54) 【発明の名称】 **l d d q異常を有する故障ブロック特定化方式**

(57) 【要約】

【課題】出力端子異常の有無に関係なく故障箇所を絞り込むことが可能となり、各種の故障モードを有する、リーク電流値の大きさに関係ない故障箇所を絞り込むことが可能とするCMOSLSIの故障箇所の特定化方式の提供。

【解決手段】基本的論理回路記（ブロック）述によるLSIの論理回路情報と、Iddq異常が発生する論理動作テストベクタ情報を用いて、各ブロック毎に演算処理を行いながら故障ブロックを絞り込み、さらに故障ブロックの位置を規格化してLSI画面上に表示する。



## 【特許請求の範囲】

【請求項1】LSIにて発生したIddq異常を有する故障ブロックを抽出する抽出手段と、抽出された前記故障ブロックの位置を表示する手段と、を有する、ことを特徴とする、Iddq異常を有する故障ブロックの特定化方式。

【請求項2】LSIの入力端子より入力するテストベクタに従って変化する、前記LSIを構成する基本的論理回路単位（ブロック）の論理動作情報と、前記LSIの入力端子に前記テストベクタを印加した際にIddq異常が検出されたテストベクタ番号情報と、を用いて、各ブロック毎に演算処理を行うことにより、故障ブロック候補を絞り込み、前記故障ブロックの位置を規格化して、前記LSI上での前記故障ブロックの位置を表示する、ことを特徴とする、Iddq異常を有する故障ブロックの特定化方式。

【請求項3】前記抽出手段が、前記LSIの入力端子より入力するテストベクタに従って変化する、前記LSIを構成する基本的論理回路単位であるブロックの論理動作情報と、前記テストベクタ毎に、LSIの論理動作の静止状態でのリーク電流であるIddqの値が予め定めた所定値を超えるテストベクタ番号を用いて所定の論理演算を行い、前記Iddq異常を有する故障ブロックを抽出する、ことを特徴とする請求項1記載の、Iddq異常を有する故障ブロックを特定化方式。

【請求項4】前記各ブロック単位の論理動作情報が、前記LSIの入力端子より入力するテストベクタに同期して変化する前記各ブロック毎の論理をシミュレーションにより抽出した、テストベクタ毎の前記各ブロックの入力の論理の組合せ、を含む、ことを特徴とする請求項2記載の、Iddq異常を有する故障ブロックの特定化方式。

【請求項5】前記ブロック毎の論理演算を行うことにより故障ブロックの抽出を行う際に、前記各ブロック毎に、Iddq異常が発生するテストベクタ番号と、Iddq異常が発生しないテストベクタ番号とにおけるテストベクタの比較において、該テストベクタ間に一致がみられないブロックを、故障ブロックとして抽出する、ことを特徴とする請求項2記載の、Iddq異常を有する故障ブロックの特定化方式。

【請求項6】前記ブロック毎の論理演算を行うことにより故障ブロックの抽出を行う際に、前記各ブロック毎に、テストベクタ毎に検出されるIddq異常が連続して発生するテストベクタ番号におけるテストベクタ群と、Iddq異常が発生しない任意の連続したテストベクタ番号における、テストベクタ間の比較において、該テストベクタ群の一致がみられないブロックを、故障ブロックとして、抽出することを特徴とする、請求項2記載のIddq異常を有する故障ブロックの特定化方式。

【請求項7】前記ブロック毎の論理演算を行うことにより故障ブロックの抽出を行う際に、前記各ブロック毎に、テストベクタ毎に検出されるIddq異常が連続して発生するテストベクタ番号におけるテストベクタ群が複数の異なったテストベクタの組合せ群として検出されたときには、Iddq異常が発生しない任意の連続したテストベクタ番号におけるテストベクタ群間の比較において、該テストベクタ群の一致がみられないブロックを、故障ブロックとして、抽出することを特徴としする、請求項2記載のIddq異常を有する故障ブロックの特定化方式。

【請求項8】前記該ブロック毎の論理演算を行うことにより故障ブロックの抽出を行う際に、組合せ回路と順序回路とを分離する、ことを特徴とする、請求項2記載のIddq異常を有する故障ブロックの特定化方式。

【請求項9】前記該故障ブロックの位置を表示する手段が、前記LSIを構成するブロックにおいて、該ブロック内の回路素子群及び該回路素子群を接続する配線をすべて囲む最小の矩形にて、予め各ブロックの原点座標を、前記LSI上に配置された各ブロックの物理座標に一致させ、抽出された前記故障ブロックの矩形を、前記LSI上の該当箇所に表示させる、ことを特徴とする請求項1記載の、Iddq異常を有する故障ブロックの特定化方式。

## 【発明の詳細な説明】

## 【0001】

【発明の属する技術分野】本発明はCMOS（相補型MOS）半導体集積回路（LSI）の故障箇所検出技術に関し、特に、静止状態電源電流の発生有無のテストベクタにおける内部回路の論理状態を比較することにより故障ブロックを抽出し、当該箇所を表示することにより故障箇所を特定化する方式に関する。

## 【0002】

【従来の技術】CAD（計算機支援型設計）を利用したシミュレーションに基づくLSIの故障箇所を絞り込むという従来の方法は、LSIの出力端子における異常の発生に関する情報をもとに、故障箇所を推定するものであった。

【0003】その第1の方法は、故障辞書（Fault Dictionary）の作成による故障シミュレーションを行うものであり、この方法はLSIの内部回路の各ブロックに故障を定義しながら、異常が発覚する出力端子、出力値、そしてテストパターン番号を、実際の故障品のデータと比較することにより、故障箇所を推定するものである。

【0004】より詳細には、図22に示すように、LSIを構成するブロックに故障を定義した回路と、正常な回路間の論理シミュレーションによる論理を比較することにより、実際の故障品での異常発生出力端子、出力値、そしてテストパターン番号の一致を比べることによ

り故障定義位置を抽出する。すなわち、図22を参照して、故障定義1502を導入したLSIの論理接続情報1503に対して、テストベクタ1504を入力して論理シミュレーション1501を行い、正常なLSIの論理接続情報1505に対して同一のテストベクタ1504を入力して論理シミュレーション1501'を行い、双方の論理シミュレーション結果を比較手段1506で比較し、不一致の場合に、故障定義1502にて定義した故障が検出される。

【0005】第2の方法は、「バックトレース方法」と称するもので、異常が発覚した出力端子、出力値、そしてテストパターン番号をもとに、出力端子から入力端子方向へ論理を逆にトレースする方法である。すなわち、LSIの入力端子より所定の信号を入力したとき、出力端子から出力する信号が期待値と異なっていたとき、その出力値と期待値との相違を利用して、出力端子側から入力端子側へ向かって内部に拡散していく信号中から故障を伝搬している信号を抽出し、故障箇所を推定し、その箇所に故障を定義して、再度、論理シミュレーションを行うことにより、実際の故障との一致を検証する方法である。

【0006】そして、LSIの故障箇所の特定化するには、LSIの複数の出力異常箇所を調査し、これらの組み合わせにより、疑似故障信号を限定しながら、故障箇所を絞り込んでいくのが一般的であった。

【0007】

【発明が解決しようとする課題】しかしながら、上記した従来の方法は、いずれも、被検出対象のLSIにおける電気回路の論理構成を理解した上でなければ、故障箇所を絞り込めないため、以下の問題があった。

【0008】まず、第1の方法である故障辞書作成により故障シミュレーション方法にて扱える故障モデルは、単一縮退故障（Stuck-at-0、Stuck-at-1）のみであり、多重縮退故障やオープン故障はシミュレーションできないため、故障モードの特定化という点からは、一般的ではなかった（すなわち検出可能な故障範囲が単一縮退故障のみとされ汎用性に乏しい）。

【0009】なぜならば、故障シミュレーションにて扱う故障は、モデル化された論理故障のみだからである。

【0010】さらに、この第1の方法においては、LSIの回路を構成するすべての信号線に対して、故障を定義していかなければならないため、膨大なデータ量となり、実用的ではなかった。

【0011】すなわち、定義する故障数（V0）はLSIを構成する回路素子数（L）の3乗から4乗に比例すると言われている。

【0012】

【数1】

$$1n(V0) \propto (3 \sim 4) \cdot 1n(L)$$

【0013】また第2の方法である、バックトレース方法は、出力端子異常の情報のみをデータとして使用する

ため、回路内部にいくつの故障が発生しているか判断することができず、従って、多重故障は扱えない。

【0014】仮に多重値が判明したとしても、出力端子での情報だけでは、どの出力情報が各故障箇所に対応するか見当がつかないため、バックトレースによる膨大な疑似故障のみが検出されることになり、絞り込みは全くの不可能とされている。

【0015】さらにバックトレース方法は、回路内部の順序回路の存在が大きな問題となっている。

【0016】論理回路は、大きく分けて、2つの回路から構成される。すなわち順序回路と組み合わせ回路である。

【0017】順序回路間に挟まれた組み合わせ回路を1つの独立した回路と考えて、その独立系の中で出力から入力側へバックトレースしながら故障を伝搬していると思われる信号を、シミュレーションにより、ある程度抽出することはできるが、順序回路はフィードバックループを考慮しなければならないため、故障を伝搬する信号を抽出することは困難である。

【0018】すなわち、順序回路において、“ある時刻における出力の論理は、それ以前の時刻に印加された入力情報に依存する回路”であるため、順序回路に入力する信号がフィードバックループとして構成されていた時、問題となる。

【0019】より詳細には、図23を参照して、例えば順序回路SC1の出力端子に出力する信号の経路は、そのタイミングを（n）とした時、順序回路SC1の入力信号としては、タイミング（n-1）における信号が入力しており、さらに、その入力信号は、フィードバックループを構成する順序回路SC1自身の出力に依存している。さらに、順序回路SC1の入力信号は、タイミング（n-2）における入力信号にも依存するという複雑な関係となる。

【0020】従って、タイミング（n）において、組み合わせ回路1中に故障を検出しても、その状態は、順序回路の入力のタイミングにあたる（n-1）におけるパターンに依存しており、さらにその状態は再度、前段の組み合わせ回路を介した順序回路自身に依存してくる。

【0021】すなわち、論理に注目したバックトレース検証は、上述したようなフィードバックの組み合わせが複数個存在した時、各順序回路間に存在する組み合わせ回路は故障の伝搬を何度も繰り返すため、故障の伝搬追跡が困難となり、現状では実用化は不可能とされている。

【0022】このため、バックトレース方法は、その手法のみで、故障箇所や故障ブロックを絞り込むことを諦め、替わってEBT（Electron Beam Tester；電子ビームテスタ）のような、物理解析方法とリンクさせた、非接触による電位コントラスト像や論理動作波形の取得により、疑似故障箇所を故障候補から消去してい

く、という方法を採らざるを得なかったというのが、実状である。

【0023】したがって、本発明は、上記事情に鑑みてなされたものであって、その目的は、CMOSLSIにおいて、出力端子異常の有無に関係なく、故障箇所を特定化することを可能とする故障箇所特定化方法を提供することにある。

【0024】また本発明は、各種の故障モードを有する、リーク電流値の大きさに関係ない故障箇所を絞り込むことを可能とし、さらに最終的に絞り込まれた故障内蔵ブロックの位置を瞬時に識別でき、故障原因を迅速に解析できるようにした故障箇所特定化方式を提供することもその目的としている。

【0025】

【課題を解決するための手段】前記目的を達成するため、本発明は、LSIにて発生したIddq異常を有する故障ブロックを抽出する抽出手段と、抽出された前記故障ブロックの位置を表示する手段と、を有する、ことを特徴とする。

【0026】

【発明の実施の形態】本発明の実施の形態について以下に説明する。まず、本発明の原理を説明する。CMOS論理回路は、回路内部に物理欠陥を有すると、一般的傾向として、“Iddq (Quiescent Vdd Supply Current)”と称する、静止状態電源電流に異常値が現れる。このIddq異常は、LSI回路内部の物理故障を外部に顕在化させるシグナル(信号)とみなすことができる。このIddqの詳細は、例えば本発明者による論文(M.Sanada, “Evaluation and Detection of CMOS-LSI with Abnormal Iddq” Microelectronics and Reliability, Vol.35, No.3, pp.619-629, 1995)の記載が参照される。本発明は、CMOS論理回路の上述した性質を利用して完成されたものである。

【0027】通常、ゲートアレイ品に代表されるASIC (Application Specific Integrated Circuits) は、予め用意された「ブロック」と称する基本的な論理を構成する回路を組み合わせるにより、所望の電気回路(電子回路)を構成することにより、実現されている。

【0028】そして、本発明は、その好ましい実施の形態において、CMOSLSIにおける故障箇所の絞り込み方法に、上述した設計方法を利用したものであり、LSIの入力端子より入力するテストベクタに従って変化する、「ブロック」と称するLSIを構成する基本的論理回路単位での論理動作情報と、そのテストベクタ毎に「Iddq」と称するLSIの論理動作の静止状態でのリーク電流の値が所定値を超えるテストベクタ番号を用いて、ブロック毎に各ブロックに、後述される演算処理を行うことにより、Iddq異常を含有する故障ブロック抽出し、該故障ブロックを、LSI全体像中の該当配

置箇所に表示するものである。

【0029】そして、上記各ブロック毎の論理動作状態は、LSIの入力端子より入力するテストベクタに同期して変化する各ブロック毎の論理をシミュレーションにより抽出した、テストベクタ毎の各ブロックの入力の論理の組合せからなる。

【0030】ブロック毎に故障ブロックを抽出する論理演算処理は、回路の種類、すなわち、組合せ回路と順序回路において異なる。

【0031】まず、組合せ回路における演算処理については、各ブロック毎に、Iddq異常が発生するテストベクタ番号と、Iddq異常が発生しないテストベクタ番号におけるテストベクタの比較において、これらのテストベクタ間に一致がみられないブロックを故障ブロックとして、抽出することにより、故障ブロックを抽出する。

【0032】順序回路における演算処理は、各ブロック毎に、テストベクタ毎に検出されるIddq異常が連続して発生するテストベクタ番号におけるテストベクタ群とIddq異常が発生しない任意の連続したテストベクタ番号におけるテストベクタ間の比較において、それらのテストベクタ群に一致がみられないブロックを故障ブロックとして抽出する。

【0033】さらに、本発明は、その好ましい実施の形態において、故障ブロックの抽出は、組合せ回路と順序回路とを分離し、大規模な回路に対しては、組合せ回路と順序回路を分離を保ちながら、階層別の分割を行うことにより、大規模から中規模、そして基本的論理回路単位にまで故障ブロック候補を絞り込んでいくものである。

【0034】次に、本発明は、その好ましい実施の形態において、故障ブロックをLSI全体像中の該当配置箇所に表示する表示方法として、LSIを構成するブロックにおいてブロック内の回路素子群、及び、これらの回路素子群を接続する配線をすべて囲む最小の矩形を、当該ブロックのサイズとして、予め定められた各ブロックの原点座標を、LSI上に配置された各ブロックの物理座標に、一致させることにより、上述した演算処理により抽出された故障ブロックを、LSI上の該当する箇所に表示させる。

【0035】

【実施例】上記した本発明の実施の形態について更に詳細に説明すべく、本発明の実施例について図面を参照して以下に説明する。

【0036】上記したように、CMOS論理回路は回路内部に物理欠陥を有すると、一般的傾向として“Iddq (Quiescent Vdd Supply Current)”と称する論理の静止状態における電源電流に異常値が現れる。

【0037】図1は、LSIにおいて物理故障の存在により貫通電流が発生する様子を模式的に示した図であ

る。すなわち、LSIの内部に物理故障が存在すると、任意のテストベクタにより設定された論理により、その物理故障を介して、又は、物理故障の影響を受けて、高位側電源V<sub>dd</sub>から低位側電源GNDへの貫通電流が発生する。

【0038】そして、この物理故障を介した論理が、テストベクタの進行と共に、出力端子へ伝搬して、出力異常が発生するモードと、出力異常が発生しないモードの、2種類のモードがある。

【0039】この違いは、図2に示すように、物理故障を介して発生するI<sub>ddq</sub>異常箇所の回路の出力論理が、スレッシュールド値(V<sub>th</sub>)に対して、高いか低いか、よるものである。すなわち、図2(b)に示すように、出力“L”の期待値に対して、I<sub>ddq</sub>異常回路(図2(c)参照)の出力(OUT)の電圧値(ロウレベル出力電圧VOL)が、スレッシュールド値(V<sub>th</sub>)よりも低ければ、正常論理として出力へ伝搬し、I<sub>ddq</sub>異常回路のロウレベル出力電圧値(VOL)がスレッシュールド値(V<sub>th</sub>)よりも高ければ、論理異常として出力端子へ伝搬し、出力異常として検出される。

【0040】また図2(a)に示すように、出力“H”期待値に対して、I<sub>ddq</sub>異常回路(図2(c)参照)の出力値(ハイレベル出力電圧VOH)が、スレッシュールド値(V<sub>th</sub>)よりも低ければ、論理異常として出力端子へ伝搬し、出力異常として検出され、I<sub>ddq</sub>異常回路のハイレベル出力電圧値(VOH)がスレッシュールド値(V<sub>th</sub>)よりも高ければ、正常論理として出力へ正常伝搬する。

【0041】本実施例においては、これら両モードに対して、有効な故障箇所の絞り込み方法を実現したものであり、このI<sub>ddq</sub>異常現象を利用して故障発生箇所を特定化する。

【0042】通常、ゲートアレイ品に代表されるASIC(Application Specific Integrated Circuits)の設計は予め準備された“ブロック”と称する基本的な論理を構成する回路を組み合わせることにより所望の電気回路を実現する設計方法である。

【0043】このようなASICのCMOS回路内部に存在する故障箇所の絞り込みは、テストベクタ毎に変化する各ブロックの論理シミュレーション情報と、I<sub>ddq</sub>異常を発生させるテストベクタ番号を用いることで可能となる。以下に詳細に説明する。

【0044】図3は、本発明の一実施例におけるI<sub>ddq</sub>異常を有する故障ブロックの表示方式を模式的に示した図である。パーソナルコンピュータ(「PC」という)又はエンジニアリングワークステーション(「EWS」という)等の計算機システム35のCRT等ディスプレイ装置の画面34上に表示されたLSI32の全体図は、以下に説明する論理情報と、テストベクタ情報と、を入力することにより、テストベクタ情報の変化に

応じて、同期して変化する、各ブロックの論理情報に沿って絞り込まれていく、疑似故障ブロックが規格化されて表示され、テストベクタの進捗とともに疑似故障ブロックの候補が絞られていく様子をモニタできる。

【0045】内部回路を構成する各ブロックの規格化表示について、図4を参照して説明する。

【0046】まずLSIを構成する各ブロックのサイズは、当該ブロックを構成する半導体素子群と、これらの素子群を接続することにより電気回路を構成する、配線の最外領域を、矩形として囲ったサイズである。そして、この規格化された各ブロックの原点は、設計方針において、任意箇所が決定されており、またLSIの設計において、各ブロックの配置情報が決定されるため、規格化された各ブロックは、LSI上に容易に割り当てることが可能である。図4(b)は、図4(a)において、ブロック内の回路素子群及びこれらの回路素子群を接続する配線をすべて囲む最小の矩形(規格化されたブロックに対応)を示している。

【0047】図5は、本発明の一実施例において、故障ブロックを絞り込む処理フローを模式的に示す図である。

【0048】図5を参照すると、LSI510の論理動作をテストするために準備されたテストベクタ501は、テストベクタ毎に変化する、LSI510を構成する各ブロック毎の論理情報とテストベクタ毎のI<sub>ddq</sub>異常の有無情報を検出するために用いられる。

【0049】まず、テストベクタ毎に変化する各ブロックの論理情報は、LSI回路検証のための論理シミュレーション502により抽出される。論理シミュレーション502はLSIの入力端子から入力するテストベクタ501に対応して出力端子に出力する期待値を検証する、論理検証のためのツールである。

【0050】論理シミュレーションによる検証過程において、同時に入力テストベクタに同期して動作するブロック毎の論理も検証されている。

【0051】従って、「ダンプ処理」と称するLSIを構成する各ブロックのテキスト名(論理シミュレーションによるシミュレーション結果情報)を指定することにより、テストベクタ毎の各ブロックの論理情報を出力することができる(図5の503参照)。

【0052】その際、LSIテスト506にて実際にLSI510にテストパターンを印加してI<sub>ddq</sub>を測定し、I<sub>ddq</sub>異常のテストパターンを検出し、この情報を基に、論理シミュレーション結果のダンプリストを抽出し、この抽出結果に対して、演算処理を施し(図5の504参照)、故障ブロックの抽出を行い(図5の505)、故障ブロックを絞り込んだ結果、表示装置の画面上にLSI上での故障ブロックの位置を表示する(図5の509参照)。なお、図5において、508の各層の外周はLSIを模式的に示しており、各層中の例えば6

×6の矩形は規格化されたブロックを表しており、これら複数の層はテストベクトル毎のLSIを模式的に表している。

【0053】従来、実行される回路検証は、LSIの内部回路の論理とともに、各ブロックから出力する論理の時間的変化をシミュレーションするのが一般的である。このため、LSIの内部回路を構成する各ブロックの入力論理はほとんど注目することがなかった。

【0054】本発明の実施例においては、故障ブロックの特定化は、LSIの内部回路を構成する各ブロックの入力論理が、テストベクトル毎に、どのように変化するかをシミュレーションし、各ブロックの入力論理情報として抽出し、後述する演算処理に用いられる。ここで、各ブロックの入力論理情報を用いる理由を、図6に示す2入力NAND回路を用いて説明する。

【0055】図6(a)を参照して、2入力NAND回路は、電源端子VDDと出力端子OUT間に並列に接続され入力IN1、IN2をゲート入力とする2つのP型MOSトランジスタPM1、PM2と、出力端子OUTと接地端子GND間に直列接続された2つのN型MOSトランジスタNM1、NM2と、を備えて構成される。図6(b)の真理値表より明らかなように、図6(a)の2入力NAND回路の入力の組合わせは4通りあるが、そのうち、3通りの入力の組合わせ、(IN1、IN2)=(H、L)、(L、H)、(L、L)は、出力がいずれも“H”で同じである。

【0056】出力の期待値をみる限り、変化はわからないが、入力をみたとき、明らかに入力値の変化に対して、内部論理が変化していることになり、各ブロックの入力論理情報が重要であることがわかる。

【0057】図7は、上述したテストベクトル毎のIddq異常有無情報を示すグラフであり、X軸はテストベクトル番号(以下「TVno.」で示す)、Y軸はIddq値を示す。正常なLSIのIddq値は規格値以下(例えば、正常状態において回路に貫通電流が発生しない時は1μA以下)であるのに対して、Iddq異常品は、規格値の数百倍から数千倍異常の貫通電流が流れる。図7では、Iddq異常のテストベクトル番号をa、b、cで示している。なお、図7においては、TVno.(a)、(b)、(c)において、同じ値のIddq異常が発生している。

【0058】次に、本発明の一実施例における、故障ブロック絞り込みのための方法について説明する。

【0059】図8は、本発明の一実施例による故障ブロックを絞り込むための原理を模式的に示した図である。図8を参照すると、複数のブロックB1、B2、B3、…、Bn、…にて構成されるLSI801の入力端子よりテストベクトル802を入力する。その入力論理は、これらのブロックで論理を展開しながら、出力端子に至る。

【0060】上述したダンプ処理(図5の503参照)により、各ブロックのテストベクトル毎の論理状態が抽出される。この様子は、図8において、各ブロック毎のダンプリスト810<sub>1</sub>～810<sub>n</sub>として示される。各ブロック毎のダンプリスト810<sub>1</sub>～810<sub>n</sub>において、TV1、TV2、…はテストベクトル番号及び該番号のベクトルを示している。

【0061】ところで、LSIにてテストベクトル毎に検出されるIddq値は、テストベクトル毎の各ブロック(B1、B2、B3、…、Bn、…)に発生するIddq値の合計であり、通常、規格値以内に収まっている。

【0062】しかしながら、物理故障を内蔵しているブロックが存在すると、そのブロックに発生する異常電流値が、LSIでのIddq値異常として検出されることになる。

【0063】上述したIddq値異常が発生しているテストベクトル番号TVno.(a)、(b)、(c)は、各ブロック毎のダンプリスト810<sub>1</sub>～810<sub>n</sub>におけるTVno.(a)、(b)、(c)にも対応している。

【0064】従って、各ブロックにおいて、Iddq異常表示されたテストベクトルと、正常状態でのテストベクトルの入力論理を、比較する(これについては後述する)ことにより、Iddq異常を内蔵するブロックを抽出することができる。

【0065】次に、本発明の一実施例において、Iddq異常を内蔵するブロックを抽出する方法について説明する。

【0066】ASICを構成するブロックは大きく2種類の回路に分類される。組合せ回路と順序回路である。このうち、組合せ回路は、そのブロックの入力端子に信号が印加されるとその論理が内部の回路を介して直接出力してくる回路形式であり、基本ゲート(AND、OR、NAND、インバータ回路等)からALU(算術論理演算ユニット)やADDER(加算)回路等の大きな規模の回路までである。

【0067】また順序回路は、クロック信号に同期してデータが一旦回路内部に蓄えられ、次のクロック信号で出力するといった出力を行う回路形式であり、フリップ・フロップ、レジスタ回路やラッチ回路等がある。本発明の一実施例においては、以下に説明するように、これら2種類の回路の故障診断は異なった方式で行われる。

【0068】まず、組合せ回路に対する故障ブロックの抽出方式について、図9から図12を参照して説明する。図9、図10は、9入力端子を有する組合せ回路においてダンプリストとして抽出された内容の一例を示す図である。簡単のために、テストベクトル番号がaという1箇所のテストベクトルにおいてのみ、Iddq異常が発生したものとする。

【0069】テストベクトル番号a(これを「TVno.(a)」と記す)における入力論理情報は(01111

0001)であり、このブロックに物理故障が内蔵しているかどうかを調査するために以下の演算を行う。

【0070】図9に示すように、TVno. (a)以外の、Iddq値が正常なテストベクタにおいて、TVno. (a)と同じ入力論理(011110001)が存在したとき、このブロックは、物理故障を内蔵していないものと判定される。なんとなれば、組合せ回路は、任意の入力論理に対して、常に、1つの内部論理しか許されないためである。従って、TVno. (a)と同じ入力論理が、正常のテストベクタにある時は、そのブロックは正常と判断される。

【0071】さらに、この判断は、ブロックが物理故障を内蔵するか、又は、内蔵しないかという選択肢に対して、TVno. (a)と同じ入力論理を有する正常のテストベクタは、正常な論理状態であるという事実が優先される、ことから導き出される。

【0072】以上から、TVno. (a)でのテストベクタにおいて、Iddq異常が発生しているが、正常なIddq値を示すテストベクタにTVno. (a)と同一のベクタがある場合、当該ブロックは物理故障を含まないものと判断される。

【0073】次に、図10に示すように、TVno. (a)以外の、Iddq値が正常なテストベクタにおいて、TVno. (a)と同じ入力論理(011110001)が存在しない時、このブロックは物理故障を内蔵しているとして抽出される。なんとなれば、上記と同様に、組合せ回路は、任意の入力論理に対して常に1つの内部論理しか許されないためであり、唯一、そのテストベクタのみが、Iddq異常を発生しており、その現象を否定する事実が存在しないからである。

【0074】同様に、同一の入力論理状態が異なるテストベクタにおいて、複数個発生した場合も、物理故障を内蔵しているかどうかの判定は、図9、図10に例示した場合と同様である。

【0075】図11、図12は、9入力端子を有する組合せ回路において、TVno. (a)、(b)、(c)という3箇所のテストベクタにおいて、Iddq異常が発生したものであり、さらにそれらの入力論理は、順に、(011110001)、(001110001)、(000110001)というように互いに異なる状態であったとする。

【0076】このブロックが物理故障を内蔵しているかどうかの調査は、基本的に、図9、及び図10を参照して説明した、単一のテストベクタでのIddq異常発生の場合と同様である。

【0077】すなわち、TVno. (a)、TVno. (b)、TVno. (c)が各々独立したテストベクタとして、それらのテストベクタと同一の入力論理が正常なテストベクタに存在するかどうかを調査することで判定する。

【0078】まず、図11に示すように、Iddq異常が発生した各テストベクタTVno. (a)、(b)、(c)の各々に対して、Iddq値が正常なテストベクタにおいて各テストベクタTVno. (a)、(b)、(c)と同じ入力論理(011110001)、(001110001)、(000110001)の少なくともどれか1つ以上存在したとき、このブロックは物理故障を内蔵していないものと判定される。なんとなれば、上記したように、組合せ回路は任意の入力論理に対して常に1つの内部論理しか許されず、従って、Iddq異常のテストベクタと同じ入力論理が正常のテストベクタにある時はそのブロックは正常と判断されるからである。

【0079】すなわち、あるブロックにおいて、3つの異なる入力論理を有するテストベクタが、1つの故障に対するIddq異常(図7の同一Iddq異常値参照)を発生させているものと仮定する。

【0080】このことは、図13に模式的に示した、テストベクタと故障箇所との関係から明らかなように、LSIにおいて3つの異なるIddq異常を発生する入力論理に対応するテストベクタTVno. (a)、(b)、(c)は、ブロックを構成する内部回路中の一箇所の故障箇所に対して、Iddq異常を発生させる、同一の論理を設定していることを意味する。なお、図13には、単一故障箇所(黒星印で示す)に対して異なった複数のテストベクタによりIddq異常が発生される際の、テストベクタとブロック内の信号伝搬経路を模式的に示している。

【0081】従って、上述した3つの入力論理のうちの少なくとも1つにおいて、正常なテストベクタの入力論理と同一のものが存在するということは、故障箇所に対して設定される論理でIddq異常が発生していないことを意味し、さらには異なる3つの入力論理に対してIddq異常を発生していないと解釈されるため、上記仮定に対して矛盾を生じる。

【0082】このため、少なくとも1つの入力論理と同一の入力論理が存在した時、このブロックは物理故障を内蔵していないと判定される。

【0083】次に、図12に示すように、Iddq値が正常なテストベクタにおいて、各テストベクタTVno. (a)、(b)、(c)のどの入力論理とも、同一の入力論理が存在しない時、このブロックは、物理故障を内蔵しているものと判定される。

【0084】さらに、このデータは、後述する、ブロック内部のトランジスタレベルの故障箇所の絞り込みにおいて大変重要なデータとなる。

【0085】次に、順序回路に対する故障ブロックの抽出方式について、図14から図17を参照して説明する。

【0086】図14、図15は、5入力端子を有する順



順序回路において抽出されたダンプリストを示している。

【0087】図14においては、TVno. (41)、(42)という連続したテストベクタにおいてIddq異常が発生している。そして、その入力論理群と同じ入力論理群が正常なテストベクタに存在する時、このブロックは物理故障を内蔵していないものと判定される。

【0088】さらに、図15に示すように、その入力論理群TVno. (41)、(42)同じ入力論理群が正常なテストベクタに存在しない時、このブロックには物理故障が内蔵されているものと判定される。

【0089】図16、図17は、5入力端子を有する順序回路において抽出されたダンプリストを示している。

【0090】図16に示すように、{TVno. (41)、(42)}、及び{TVno. (51)、(52)、(53)、(54)}という連続した2箇所のテストベクタにおいて、Iddq異常が発生している。これらの入力論理群の1つ以上に対して、同じ入力論理群が正常なIddq値を示すテストベクタに存在する時(Iddq異常のテストベクタ51~54と同一の入力論理のテストベクタ102~105はIddq正常)、このブロックは物理故障を内蔵していないものと判定される。

【0091】さらに、図17に示すように、これらの入力論理群と同じ入力論理群が正常なテストベクタに存在しない時、このブロックは物理故障が内蔵していると判定される。

【0092】この判断理由を、図18に示した、基本的な順序回路であるD型フリップ・フロップ(「D型F/F」を略記する)を参照して以下に説明する。

【0093】図18は、1個のインバータ回路、2個の2入力AND回路、及び、2個の2入力NOR回路から構成される2入力及び2出力の端子群を有するD型フリップフロップの構成を示している。Dはデータ端子、CLKはクロック端子、Q、Q\*は出力端子、相補出力端子を示している。図19は、図18に示したD型フリップフロップの真理値表を示す。

【0094】このD型F/Fにおいて、Q出力側2入力NOR回路NOR1が“H”出力になった時、Iddq異常が発生したとすると、この異常は、図19に示す論理テーブル表のテストベクタTVno. (6)、(7)、(10)、(11)、(12)、(13)においてIddq異常が発生する。

【0095】このうち、TVno. (7)、(11)、(13)に注目したとき、同様な入力論理が、TVno. (3)、(5)、(9)においても設定されている。しかしながら、これらTVno. (3)、(5)、(9)の入力論理においては、Iddq異常は発生していない。

【0096】この理由として、順序回路は、クロック信号に同期して、データが一旦回路内部に蓄えられ、次のクロック信号で出力するという回路形式のためである。

【0097】すなわち、図19に示す論理テーブル表から明らかなように、テストベクタTVno. (7)、(11)、(13)において保持されている論理は、正転出力Q側の2入力NOR回路NOR1が“H”、及び反転出力Q\*側の2入力NOR回路NOR2が“L”出力となった論理であるのに対し、テストベクタ(3)、(5)、(9)において保持されている論理は、正転出力Q側の2入力NOR回路NOR1が“L”、及び反転出力Q\*側2入力NOR回路NOR2が“H”出力となった論理であるためである。

【0098】従って、Q出力側2入力NOR回路NOR1が“H”出力になった時、Iddq異常が発生する状態は、テストベクタTVno. (7)、(11)、(13)での保持状態で異常として、検出されている。

【0099】このように、順序回路における故障箇所検出の方式はD型F/Fへの論理の設定と保持状態の入力論理を、1つの組合せ群テストベクタ{TVno.

(6)、(7)}及び{TVno. (10)、(11)、(12)、(13)}として調査しなければならない。すなわち、順序回路の故障箇所検出の方式は以前の入力論理との組合せの群が正常なテストベクタでの入力論理の中に存在するかどうか調査しなければ、そのブロックに故障が存在するかどうかの判定がなされないことになる。

【0100】図20は、本発明の一実施例において、LSIを、階層別に分割し、故障ブロックを絞り込む方式を模式的に示す図である。

【0101】階層別ブロック構成は、設計時に使用される「ライブラリ」と称する、予め用意されている、基本的な回路構成されたブロックの単位で、解析を行う方式が一般的であるが、大規模化されたLSIにおいては、膨大なブロック数となることが予測される。

【0102】従って、任意の大きさに再分割して、LSI内部のブロック構成を変える必要がある。その際、ブロック構成で注意すべきことは、1つの階層単位の中に組合せ回路と順序回路を区別して階層分割しなければならないことである。

【0103】その理由は、上述した通り、組合せ回路と順序回路での演算処理の方式が異なるためである。

【0104】図20を参照すると、階層分割Aにおいて組合せ回路と順序回路が混在しているため、順序回路を境にして、組合せ回路を、分割しやすい階層構成(a1、a2、a3)として故障箇所の絞り込みを行っている。

【0105】次に、故障が発生しているとして抽出された階層ブロックa1において、階層ブロックa1を構成する階層構成(b1、b2、b3)にて、故障箇所の絞り込みを行う。

【0106】そして最後に、最小単位である、「基本的回路構成」であるブロックb2を抽出する。



【0107】以上説明したように、本発明の一実施例においては、故障ブロックの抽出はテストベクタ毎のLSIを構成している「ブロック」と称する基本的な論理を有する回路単位の入力論理の変化情報と、Iddq異常を有するLSIのIddq異常有無のテストベクタ番号情報から、上記した演算処理により、故障ブロックを抽出することができる。

【0108】本発明の一実施例における、故障ブロックの抽出方式において、効率的な方法は、図21に示すように、Iddq異常が発覚している全テストベクタまたは一部のテストベクタでの、各ブロックの入力論理情報から、これらのテストベクタに共通の入力論理を有する疑似ブロックを抽出し、次にIddqが正常なテストベクタでの各ブロックの入力論理と共通な入力論理を有するブロックを除去することにより、迅速な故障ブロック特定の実現が可能となる。この様子は、上述した、表示装置におけるLSI上での障箇ブロックの位置表示方式（図5参照）により、画面上でモニタできる。

【0109】

【発明の効果】以上説明したように、本発明によれば、CMOSLSIの故障箇所の特定に際して、Iddq異常が発生したという現象を用いて、直接、故障箇所候補を絞り込むようにしたものであり、この方法は、以下に示す6つの大きな効果を奏する。

【0110】（1）第1の効果は、出力端子異常の有無に関係なく、故障箇所を絞り込める、ということである。これは本発明の最も顕著な効果である。

【0111】（2）第2の効果は、容易に故障箇所を絞り込むことを可能とする、ということである。

【0112】すなわち、本発明によれば、LSIの故障ブロックの特定に際して、LSI設計段階での検証ツールとして用いる論理シミュレーションを基にした各ブロック毎のダンプリストと、Iddq異常が発生したテストベクタ番号のみのデータを用意すればよいから、回路が解らなくても、簡単に故障箇所を絞り込むことができる。

【0113】さらに、上述したデータは、故障品のIddq異常が発生したテストベクタ番号のみでよいから、実際の故障品がなくても、故障の解析が可能であるという利点も有している。

【0114】（3）第3の効果は、単一縮退故障と共に、多重縮退故障や、オープン故障等の物理故障を検出することができる、ということである。

【0115】多重故障である複数のIddq異常が発生した場合、まず各故障箇所に対して流れる貫通電流は一定であるため、テストベクタに対するIddq値を読みとるだけで、何箇の故障箇所が発生しているかの識別ができ、さらに、その各々に対して演算処理が可能だから、容易に多重故障を検出できる。

【0116】さらに、Iddq異常として発覚するオー

ブン故障は、その箇所の論理が、オープン故障を介して、貫通電流を発生し、本発明においては、Iddq異常の発生の有無は、印加される論理に依存するため、上述したように、容易に解析できるためである。

【0117】（4）第4の効果は、処理の高速化が可能である、ということである。本発明の方法は、コンピュータが得意とする演算処理のみで行えるため、高速に処理が可能である。また、LSIが大規模になったとしても、LSIを分割したブロック単位での演算が可能とされており、コンピュータ容量の影響を受けないという利点を有している。

【0118】（5）第5の効果は、通常のLSIの解析で問題となる、順序回路のフィードバックループが、本発明においては、全く問題がないという、ことである。

【0119】すなわち、本発明においては、Iddq異常が発生したという現象のみから、基本的論理回路単位の検索を行うことにより、自動的に故障箇所を絞り込むを行うことが可能とされているため、順序回路のフィードバックループや、加算器の繰返しループは、本発明においては、特に問題とする必要がないためである。

【0120】（6）第6の効果は、本発明の適用は、Iddq異常値の大きさに関係しない、ということである。

【0121】すなわち、本発明において、故障解析に必要なデータは、Iddq異常が発生したテストベクタ番号であり、Iddq異常値の大きさには関係しないため、正常品とわずかな差の現れるサンプルにおいても、その発生箇所を絞り込むことが可能となる。

【0122】（7）第7の効果は、順序回路において、Iddq異常の検出による故障箇所の特定化は、異常が発生したテストベクタを抽出すればよく、さらに、内部の論理設定が以前のテストベクタに依存している時には、そのテストベクタとIddq異常のテストベクタでの入力論理の組合せを単位とし、その組合せの有無を他のテストベクタでの論理の組合せから探索すればよいから、従来の出力端子異常から出発する、バックトレースによる故障箇所の絞り込み方式のように、論理の繰返しを心配する必要はまったくない。

【0123】（8）第8の効果は、故障箇所を内蔵したブロックを抽出する演算処理は、各ブロックの規格化表示により、テストベクタの変化に従って、逐次モニタすることができる、ということである。

【0124】さらに最終的に絞り込まれた故障内蔵ブロックの位置を瞬時に識別でき、そのデータを故障解析装置に転送することにより、ナビゲーションを瞬時に行え、故障原因を迅速に解析できる。

【0125】（9）第9の効果は、故障ブロック特定は、完全なソフトウェアによる演算処理にて実行でき、故障サンプルを、直接、使用しない。本発明においては、サンプルは、故障解析装置に装着しておき、結果が

出力されると、その座標を転送し、あるいは解析装置に  
入力して、迅速に故障原因の絞り込みを行うことができ  
る、ということである。

【図面の簡単な説明】

【図1】本発明の一実施例を説明するための図であり、  
物理故障の存在による貫通電流発生の様子を模式的に示  
す図である。

【図2】本発明の一実施例を説明するための図であり、  
物理故障を介して発生するIddq異常箇所がLSIの  
出力端子へ及ぼす影響を模式的に示す図である。

【図3】本発明の一実施例における、Iddq異常を有  
する故障ブロックの表示方式を示す概略図である。

【図4】本発明の一実施例を説明するための図であり、  
内部回路を構成する各ブロックの規格化表示（ブロック  
と最小矩形）を模式的に説明するための図である。

【図5】本発明の一実施例における、故障ブロックを絞  
り込む処理フローを模式的に示す図である。

【図6】本発明の一実施例を説明するための図である。

【図7】本発明の一実施例を説明するための図でありテ  
ストベクタ番号に対してIddq異常が発生する状態を  
示すグラフである。

【図8】本発明の一実施例における、故障ブロックを絞  
り込むための原理を模式的に示す図である。

【図9】本発明の一実施例において、組合せ回路に対す  
る故障ブロックの抽出方式を説明するための図であり、  
9入力端子を有する組合せ回路にて1箇所のテストベク  
タにてIddq異常が発生した場合のダンプリストの一例  
を示す図であり、ブロックが故障を内蔵していない例  
を説明するための図である。

【図10】本発明の一実施例において、組合せ回路に対  
する故障ブロックの抽出方式を説明するための図であ  
り、9入力端子を有する組合せ回路にて1箇所のテスト  
ベクタにてIddq異常が発生した場合のダンプリスト  
の一例を示す図であり、ブロックが故障を内蔵している  
例を説明するための図である。

【図11】本発明の一実施例において、組合せ回路に対  
する故障ブロックの抽出方式を説明する図であり、入力  
端子を有する組合せ回路にて3箇所の異なるテストベク  
タにてIddq異常が発生した場合のダンプリストの一例  
を示す図であり、ブロックが故障を内蔵していない例  
を説明するための図である。

【図12】本発明の一実施例において、組合せ回路に対  
する故障ブロックの抽出方式を説明するための図であ  
り、9入力端子を有する組合せ回路にて3箇所の異なる  
テストベクタにてIddq異常が発生した場合のダンブ  
リストの一例を示す図であり、ブロックが故障を内蔵し  
ている例を説明するための図である。

【図13】本発明の一実施例を説明するための図であ  
り、単一故障が、3箇所の異なるテストベクタにてIddq異常として検出されている例に対する、故障発生現

象を説明するための図である。

【図14】本発明の一実施例において、順序回路に対す  
る故障ブロックの抽出方式を説明するための図であり、  
5入力端子を有する順序回路にて連続した2個のテスト  
ベクタにてIddq異常が発生した場合のダンプリスト  
の一例を示す図であり、ブロックが故障を内蔵していな  
い例を説明するための図である。

【図15】本発明の一実施例において、順序回路に対す  
る故障ブロックの抽出方式を説明する図であり、5入力  
端子を有する順序回路にて連続した2個のテストベクタ  
にてIddq異常が発生した場合のダンプリストの一例  
を示す図であり、ブロックが故障を内蔵している例を説  
明するための図である。

【図16】本発明の一実施例において、順序回路に対す  
る故障ブロックの抽出方式を説明するための図であり、  
5入力端子を有する順序回路にて連続した2箇所の異な  
ったテストベクタ群にてIddq異常が発生した場合の  
ダンプリストの一例を示す図であり、ブロックが故障を  
内蔵していない例を説明するための図である。

【図17】本発明の一実施例において、順序回路に対す  
る故障ブロックの抽出方式を説明するための図であり、  
5入力端子を有する順序回路にて連続した2箇所の異な  
ったテストベクタ群にてIddq異常が発生した場合の  
ダンプリストの一例を示す図であり、ブロックが故障を  
内蔵している例を説明するための図である。

【図18】本発明の一実施例を説明するための図であ  
り、2入力D型フリップ・フロップの回路構成を示す図  
である。

【図19】本発明の一実施例を説明するための図であ  
り、2入力D型フリップ・フロップにて代表した順序回  
路の故障箇所を絞り込みの特徴を説明する図であり、D  
型フリップ・フロップに入力したテストベクタに対する  
出力論理のテーブルである。

【図20】本発明の一実施例において、LSIを階層構  
造別に分割しながら、故障箇所を絞り込んでいく様子を  
模式的に示す図である。

【図21】本発明の一実施例を説明するための図であ  
り、Iddq異常が発覚しているテストベクタ群の共通  
の入力論理を有する疑似ブロックを、Iddqが正常な  
テストベクタ群での入力論理と共通な入力論理を有する  
ブロックを除去する効率的な抽出方式を説明するための  
図である。

【図22】従来の故障辞書作成による故障シミュレーシ  
ョン方法を説明するための図である。

【図23】順序回路に入力する信号がフィードバックル  
ープとなる回路構成の一例を説明するための図である。

【符号の説明】

501 テストパターン

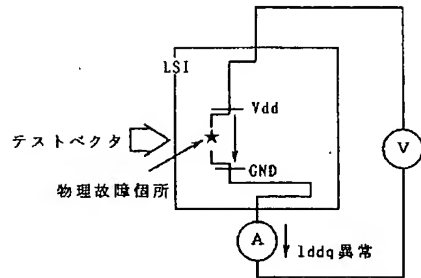
502 論理シミュレーション

503 ダンプリスト抽出

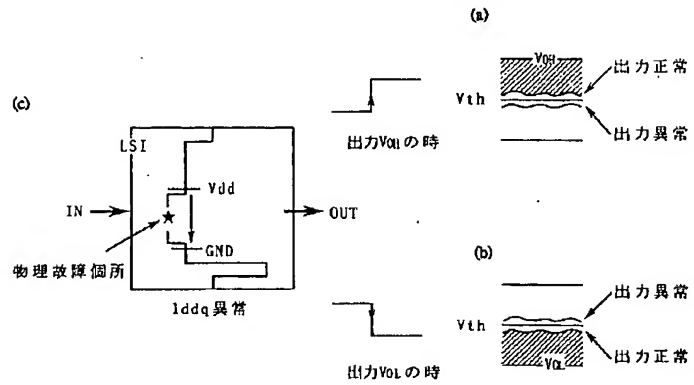
504 演算処理  
505 故障ブロック抽出  
506 LSIテスト

507  $I_{ddq}$ 異常テストパターン検出  
508 LSI上での故障ブロックの位置表示  
510 LSI

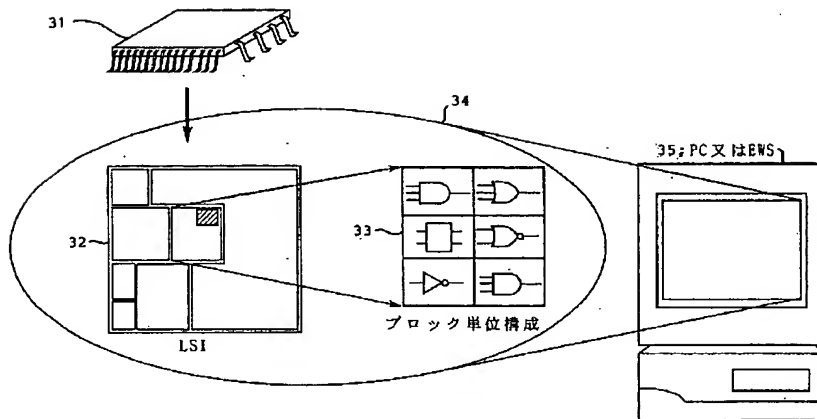
【図1】



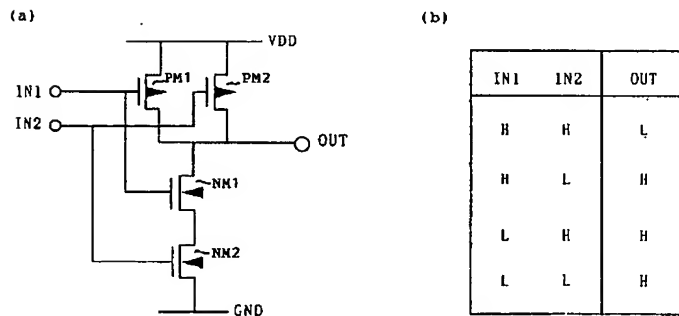
【図2】



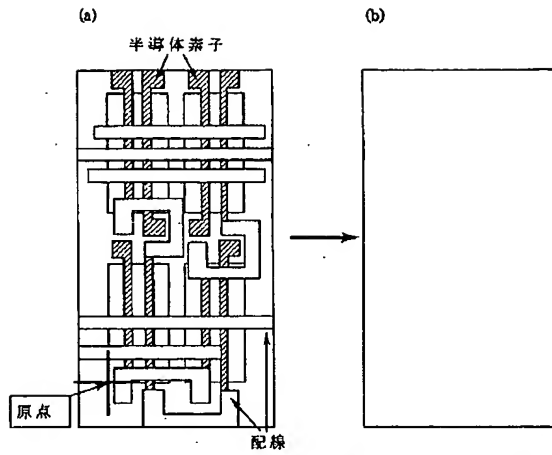
【図3】



【図6】

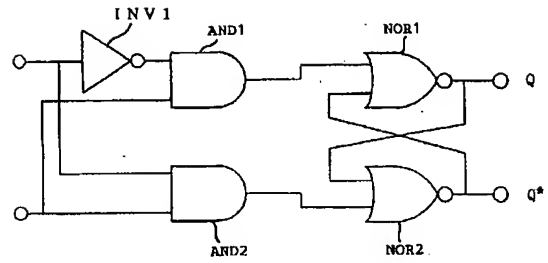


【図4】

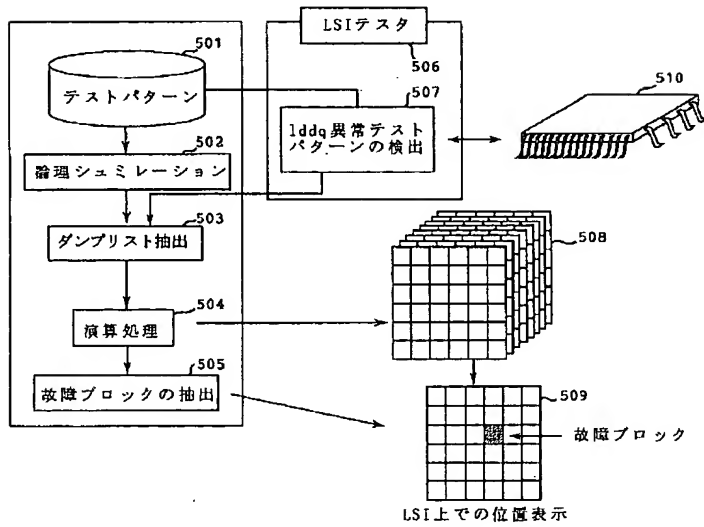


ブロックの最小矩形: ブロック内の回路素子群及びそれらの回路素子群を接続する配線をすべて囲む最小の矩形

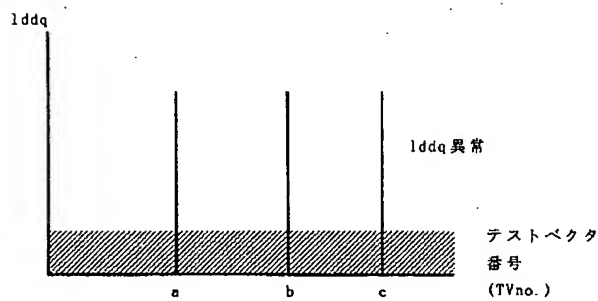
【図18】



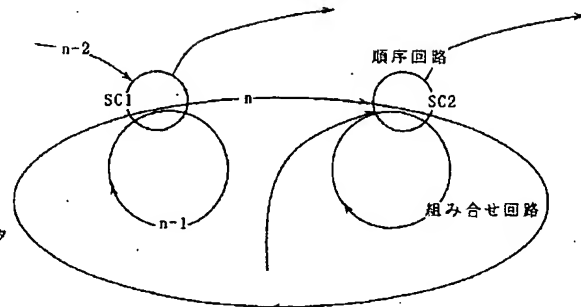
【図5】



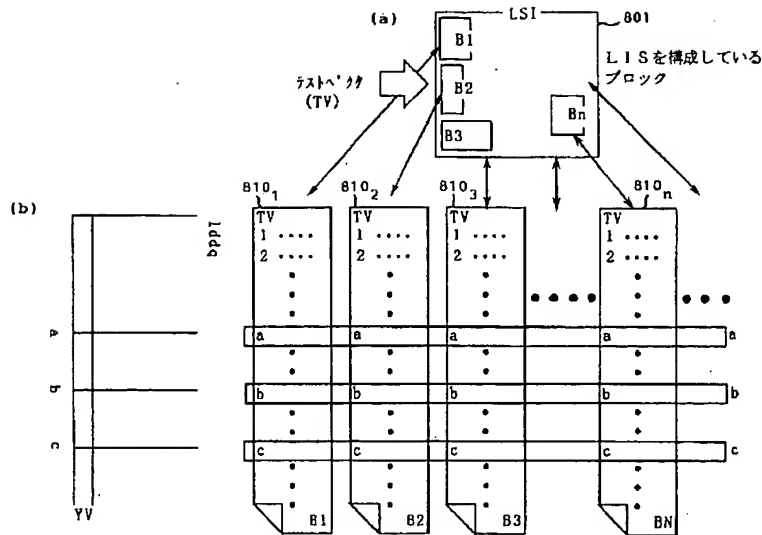
【図7】



【図23】



【図8】



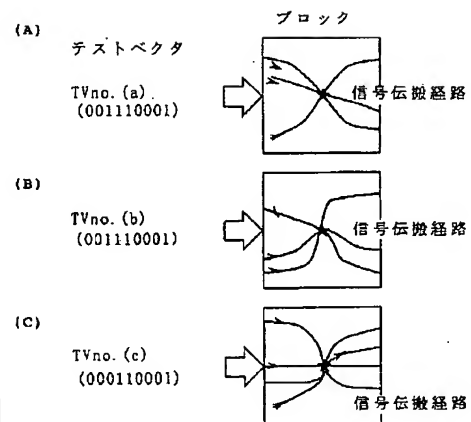
【図9】

テストベクタ 番号	NNNNNNNNNN	
	000000000	
	123456789	
1	000001000	正常
2	011001000	
3	011001001	
...	...	
a	011110001	(lddq異常)
...	...	
	011110001	正常

TVno. (a) でのテストベクタにおいて lddq 異常が発生しているが  
正常な lddq 値を示すテストベクタに TVno. (a) と同一のベクタあり。

本ブロックは物理故障を内蔵していない

【図13】



【図10】

テストベクタ 番号	NNNNNNNNNN	
	000000000	
	123456789	
1	000001000	正常
2	011001000	
3	011001001	
⋮	⋮	
⋮	⋮	
⋮	⋮	
⋮	⋮	
a	011110001	(lddq異常)
⋮	⋮	
⋮	⋮	
⋮	⋮	正常

TVno. (a) でのテストベクタにおいてlddq異常が発生しているが  
正常なlddq値を示すテストベクタにTVno. (a) と同一のベクタなし。

↓  
本ブロックは物理故障を内蔵している

【図11】

テストベクタ 番号	NNNNNNNNNN	
	000000000	
	123456789	
1	000001000	正常
2	011001000	
3	011001001	
⋮	⋮	
⋮	⋮	
⋮	⋮	
a	011110001	(lddq異常)
⋮	⋮	
b	001110001	(lddq異常)
⋮	⋮	
c	000110001	(lddq異常)
⋮	⋮	
⋮	⋮	
⋮	001110001	正常 TVno. (b) と同一

TVno. (a) (b) (c) でのテストベクタにおいてlddq異常が発生しているが正常な  
lddq値を示すテストベクタにlddq異常のベクタと同一のベクタ(b)あり。

↓  
本ブロックは物理故障を内蔵していない

【図12】

テストベクタ 番号	NNNNNNNNNN	
	000000000	
	123456789	
1	000001000	正常
2	011001000	
3	011001001	
⋮	⋮	
⋮	⋮	
a	011110001	(lddq異常)
⋮	⋮	
b	001110001	(lddq異常)
⋮	⋮	
c	000110001	(lddq異常)
⋮	⋮	
⋮	⋮	
⋮	⋮	正常

TVno. (a) (b) (c) でのテストベクタにおいてlddq異常が発生しているが正常なlddq値を示すテストベクタにlddq異常のベクタと同一のベクタなし。

本ブロックは物理故障を内蔵している

【図14】

テストベクタ 番号	NNNNN	
	00000	
	12345	
1	00000	正常
2	01101	
3	01100	
⋮	⋮	
⋮	⋮	
41	01111	(lddq異常)
42	01110	
⋮	⋮	
⋮	⋮	
87	01111	正常
88	01110	
⋮	⋮	
⋮	⋮	
96	01110	

lddq異常が発生している連続したテストベクタTVno. (41), (42), と同一の連続ベクタTVno. (87), (88) にて正常なlddq値を示すテストベクタあり。

本ブロックは物理故障を内蔵していない



【図15】

テストベクタ 番号	NNNNN 12345	
1	00000	正常
2	01101	
3	01100	
⋮	⋮	
41	01111	←(lddq異常)
42	01110	
⋮	⋮	正常
87	01100	
88	01110	
⋮	⋮	

lddq異常が発生している連続したテストベクタTVno. (41), (42), と同一の連続ベクタが正常なlddq値を示すテストベクタに存在しない。

↓

本ブロックは物理故障を内蔵している

【図16】

テストベクタ 番号	NNNNN 12345	
1	00000	正常
2	01101	
3	01100	
⋮	⋮	
41	01111	←(lddq異常)
42	01110	
⋮	⋮	正常
51	01110	
52	01100	
53	01101	
54	10000	
⋮	⋮	正常
102	01110	
103	01100	
104	01101	
105	10000	

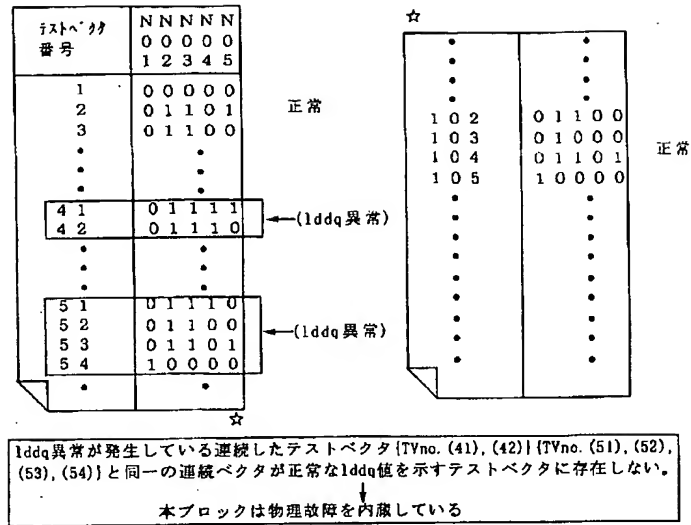
☆

lddq異常が発生している連続したテストベクタTVno. (41), (42) TVno. (51), (52), (53), (54) と同一の連続ベクタが正常なlddq値を示すテストベクタに存在する。

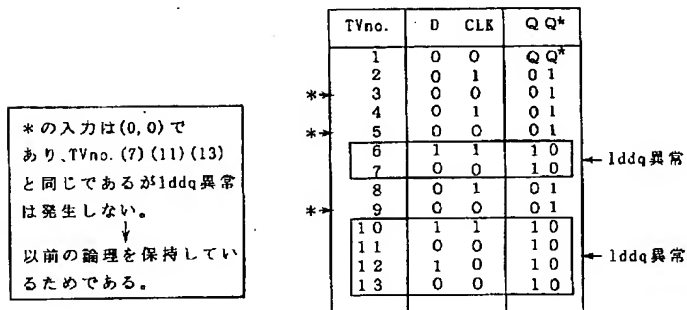
↓

本ブロックは物理故障を内蔵していない

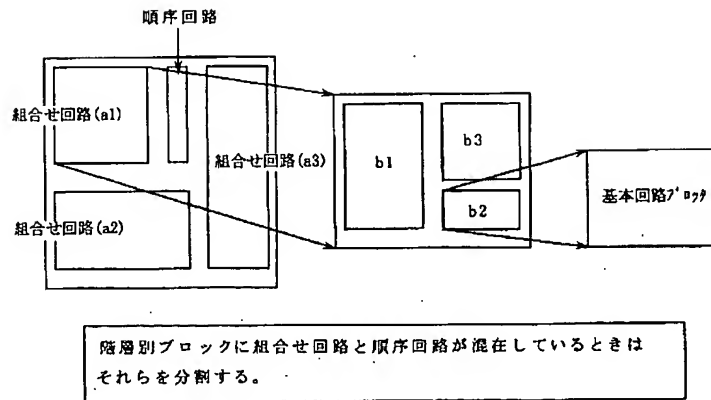
【図17】



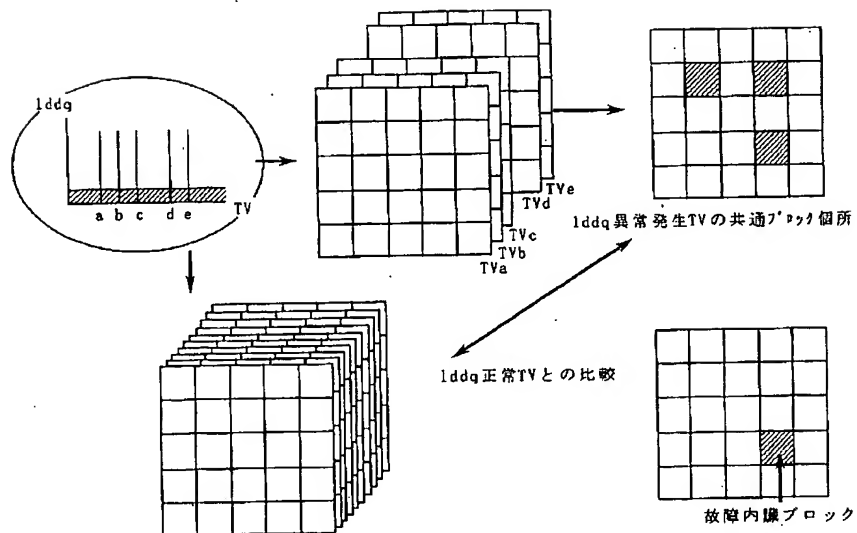
【図19】



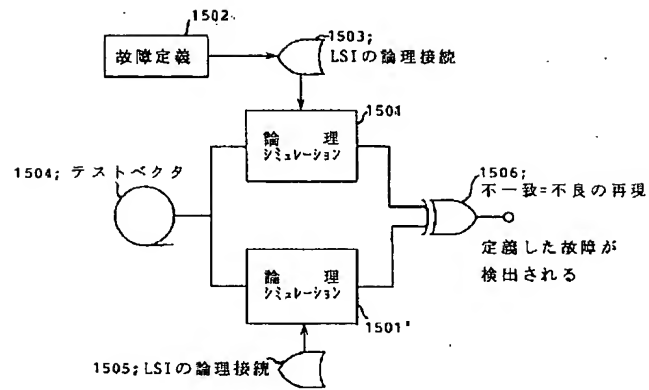
【図20】



【図21】



【図22】



**This Page is Inserted by IFW Indexing and Scanning  
Operations and is not part of the Official Record.**

## **BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

☐ **BLACK BORDERS**

☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**

☐ **FADED TEXT OR DRAWING**

☒ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**

☐ **SKEWED/SLANTED IMAGES**

☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**

☐ **GRAY SCALE DOCUMENTS**

☐ **LINES OR MARKS ON ORIGINAL DOCUMENT**

☐ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**

☐ **OTHER:** \_\_\_\_\_

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning these documents will not correct the image  
problems checked, please do not report these problems to  
the IFW Image Problem Mailbox.**